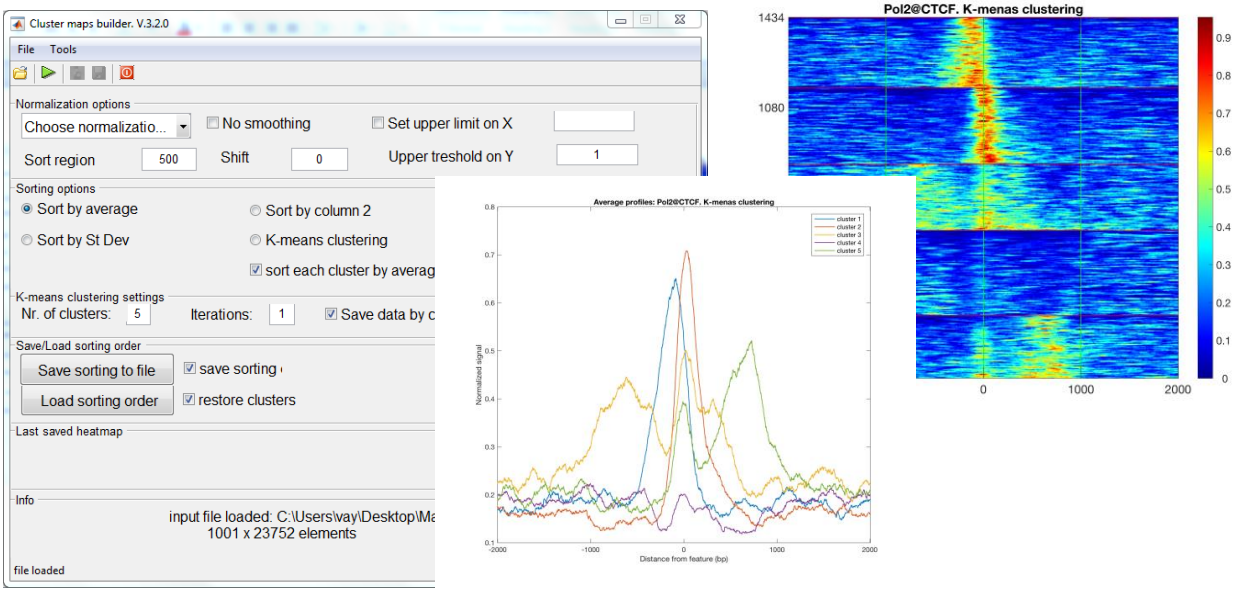


NucTools: Cluster maps builder

User manual

** Yevhen Vainshtein [1], Vladimir B. Teif [2]*

[1] *Fraunhofer-Institut für Grenzflächen- und Bioverfahrenstechnik IGB, Nobelstraße 12, 70569 Stuttgart, Germany;*

[2] *School of Biological Sciences, University of Essex, Wivenhoe Park, CO4 3SQ Colchester, UK*

**Correspondence* yevhen.vainshtein@igb.fraunhofer.de

Table of Contents

Disclaimer

The Cluster maps builder tool (CMB) is a part of NucTools and at the moment at constant development and therefore may show some instability and bugs.
Some known features are addressed here. In the case you are facing new bug please feel free to contact me: y.vainshtein(at)zmbh.uni-heidelberg.de

Introduction

The "Cluster maps builder" (CMB) is primarily designed to visualize nucleosome occupancy profile of thousands of features aligned at genomic coordinate corresponding to a specific feature, like transcription factor binding site or transcription/translation initiation or termination site, using a heatmap representation. The CMB includes a K-means clustering step and is able to propagate sorting/clustering order from initial matrix to a different matrix of the same size and dimensions.

The CMB is written on MATLAB and is using a MATLAB Java-based GUI (GUIDE). In the moment, we distribute this application as a package of MATLAB and Perl scripts and therefore the prerequisite of CMB's usage is availability of MATLAB installation.
The initial development was done on MATLAB 2014b but the program was tested for compatibility with 2015a/b and the latest 2016a. The CMB is working both on Window and MacOS X. It was not yet tested on a native Linux operating system.

Package content

The CMB package consists of the following scripts:

Script name	Description
heatmap_builder.m	Main script of a CMB package, containing all functions evaluating interface calls and performing calculations. *Copyright Yevhen Vainshtein, Vladimir Teif*
heatmap_builder.fig	GUIED user interface

Scripts published at MathWorks file exchange:

Script name	Description
heatmap.m	Displays a matrix as a heatmap image *Copyright 2014 The MathWorks, Inc.*

nanmean.m	Returns the sample mean of X, treating NAs as missing values *Copyright 1993-2004 The MathWorks, Inc*
smoothc.m	Smooths a 2D matrix using a cosine taper function. *Author: Linda Winkler*
progressbar.m	progressbar provides an indication of the progress of some task using graphics and text *Author: Steve Hoelzer*
statusbar.m	statusbar set/get the status-bar of Matlab desktop or a figure *Author: Yair M. Altman*

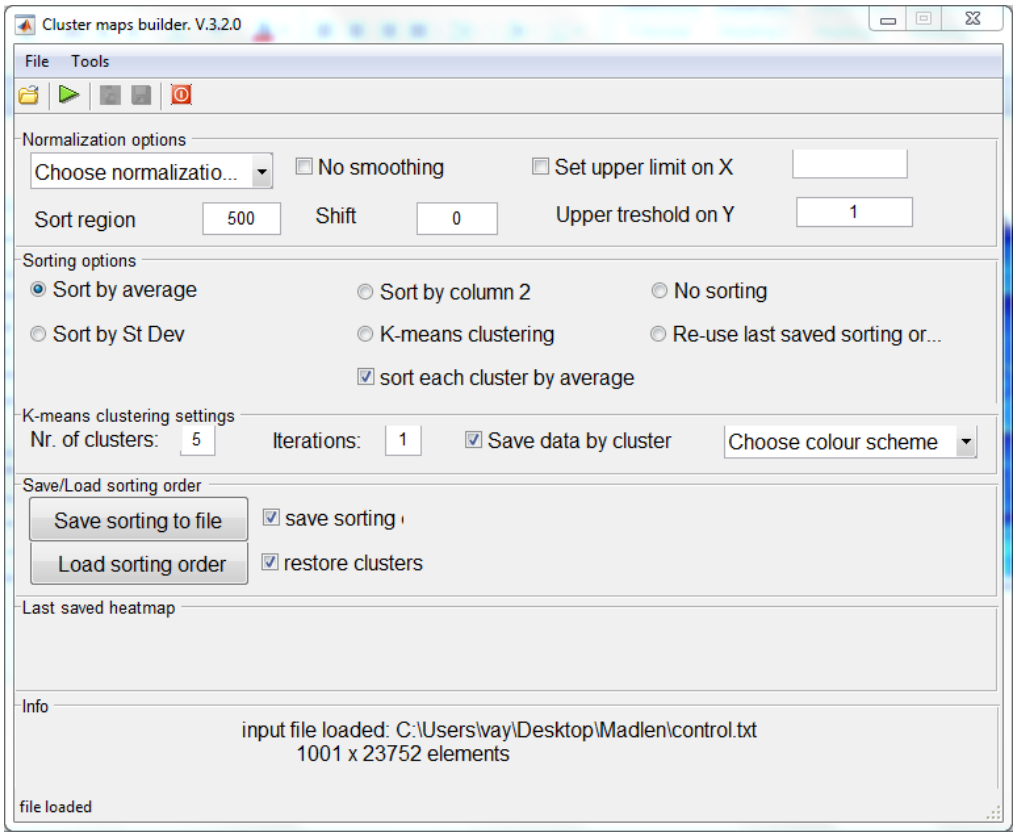
Perl scripts (in MacOs X version)

Script name	Description
countlines.pl	Return Nr. of lines of the input text table
file_size.pl	Return file size in MBs

Additional files (MATLAB variable storage files):

MyBlueColormap.mat clusters_order.mat
MyRedColormap.mat sorting_order.mat

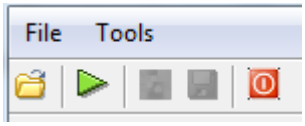
Cluster maps builder GUI



The GUI consists of 7 major panels (from top to bottom), and a status bar:

- Menu panel
- Top buttons panel
- Normalization options panel
- Sorting options panel
- K-means clustering settings panel
- Sorting order backup panel
- Info panel

Menu & Buttons panels



"File menu > Open matrix (Ctrl+O)" – opens standard system "Open file" dialog.

The proper input file for the CMB application is a tab-delimited text file, containing normalized occupancy values for features aligned at defined genomic region (the output of **aggregate_profile.pl** script from NucTools package).

CMB accepts any tab-delimited text file without o with a header of the following type:

		Distance to a feature start or center (bp)						
Feature ID	**Sorting order (expression)**	**-100**	**-99**	**...**	**0**	**...**	**+99**	**+100**
column 1	**column 2**	**column 3**	**column 4**					
ID1	-10.98	3	2.008		0.00012		0.22	0.45
ID2	0.8765	1.9018	1.022		0.001		0.00012	0
...								

Note:
Column two is an option. One can use it, for example, to provide expression values or any other arbitrary score. These values can be used to sort data matrix accordingly for heatmap representation.

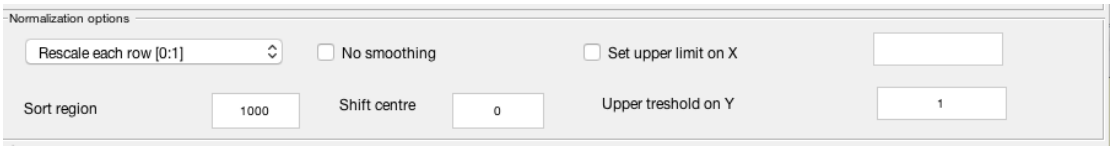
"Tools menu > Run (Ctrl+R)" – Normalize, rescale, sort or perform K-means clustering of the data using analysis settings defined in corresponding panels below and draw a cumulative occupancy profile using mean of all values in each column after data normalization or rescaling (for details see below in the section "normalization options")

"Tools menu > Visualize (Ctrl+D)" – Draw a heatmap visualization of occupancy matrix. When prompted, specify a heat map name. It will be used as an image title as well as an image file name.
Heatmap and corresponding aggregate profiles (in the case of a K-mean clustering analysis option) figures will be saved automatically in the same folder as the input matrix.

"File menu > Save clustered tables (Ctrl+S)" – opens standard system "Save file as" dialog. The program will save a file containing Feature ID, mean of the occupancy in the sort region and cluster ID if applicable. As well, the original matrix with features aligned at specific genomic regions will be saved according to clusters and sorting.
Such tables could be used again with the CMD to perform further analysis of selected clusters.

"File menu > Exit (Ctrl+X)" – exit CBMT program without saving unsaved data.

Normalization options panel



The "Normalization options" panel allows changing analysis settings related to the initial data treatment before sorting or K-means clustering.

The "**Choose normalization method**" drop-down menu contains following options:

- "**Rescale complete matrix [0:1]**" – Finds a global minimum and global maximum among all values in the matrix and assign it to 0 and 1 correspondingly. Assign the value for all occupancy values in the matrix from the range [0:1]

$$New.Occupancy_{xy} = \frac{Old.Occupancy_{xy} - min(\square\square\square\square\square\square)}{max(Matrix) - min(Matrix)}$$

  Where *Matrix* is a complete data array, *X* and *Y* rows and columns indexes, $Occupancy_{xy}$ is a nucleosome occupancy of feature $FeatureID_y$ at the coordinate X.

- "**Rescale each row [0:1]**" – Finds a minimum and maximum among all values in the each row and assign it to 0 and 1 correspondingly. Rescale all values in the row *Y* from 0 to 1:

$$New.Occupancy_{xy} = \frac{Old.Occupancy_{xy} - min(row_y)}{max(row_y) - min(row_y)}$$

  where row_y is a vector of occupancy values of a feature $FeatureID_y$

- "**Normalize each row to a maximum**" – divide the occupancy value in the row *Y* by the maximum value among all values in the each row:

$$New.Occupancy_{xy} = \frac{Old.Occupancy_{xy}}{max(row_y)}$$

- "**Normalize each row to a global maximum**" – divide the occupancy value in the row *Y* by the maximum value among all values in the whole matrix:

$$New.Occupancy_{xy} = \frac{Old.Occupancy_{xy}}{max(Matrix)}$$

- "**Normalize each row to a leftmost value**" – divide the occupancy value in the row *Y* by the leftmost occupancy value from the sort region for each row *Y*:

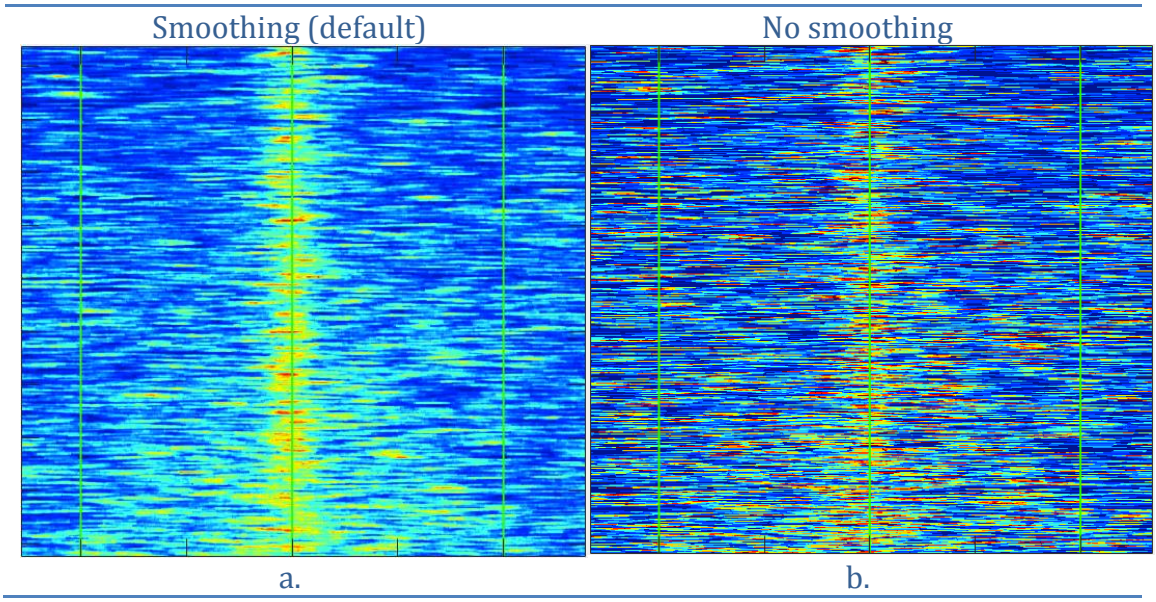
$$New.Occupancy_{xy} = \frac{Old.Occupancy_{xy}}{Old.Occupancy_{1y}}$$

- "**No normalization; Remove values above the threshold**" – read the value from "Upper threshold on Y" and replace all occupancy values above it with the threshold value (for example, remove outliers caused by piling-up of to many reads due to reads mapping artifacts)

- "**No normalization**" – process data without any prior normalization.

The rest of options in the "Normalization options" panel can be divided into two categories: analysis settings and visualization settings.

Visualization settings: "no smoothing"
By default the "no smoothing" checkbox is deactivated and before plotting matrix on a heatmap, the 2D matrix is smoothed using a cosine taper function for better visualization:

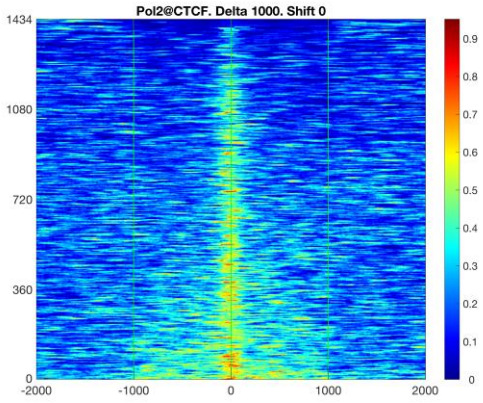


Visualization settings: "Set upper limit on X"
Limit the X axis when drawing average aggregated profile and per-cluster aggregated profiles. The data matrix itself is not change and the heatmap visualization will be done for whole data set.

Note: the limitation on X could be specified in the text field on the left from checkbox

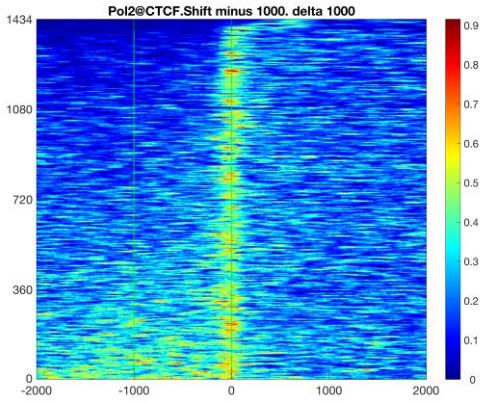
Analysis settings: "sort region" and "shift center"
These two options are a key to specify the coordinates for further analysis, relative to the genomic feature. By default we assume the data is aligned and centered at the middle of TF binding site. The original data matrix could be spanning from several kbs downstream to several kbs upstream from the genomic region (the limitation is only due to the computer RAM and CPU). But we can limit further analysis only to the region centered at 0+shift and spanning from minus "sort region" to plus "sort region" to focus only on specific data range.
Such data range is indicated on the heatmap with vertical lines at the center and at boundaries.

shift=0, sort region=1000



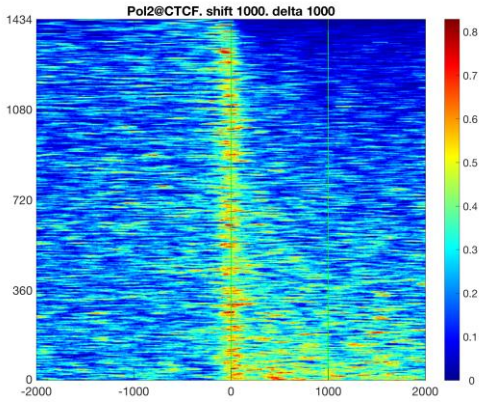
a.

shift=-1000, sort region=1000



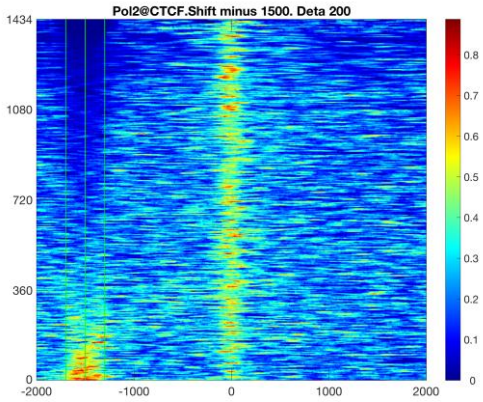
b.

shift=1000, sort region=1000



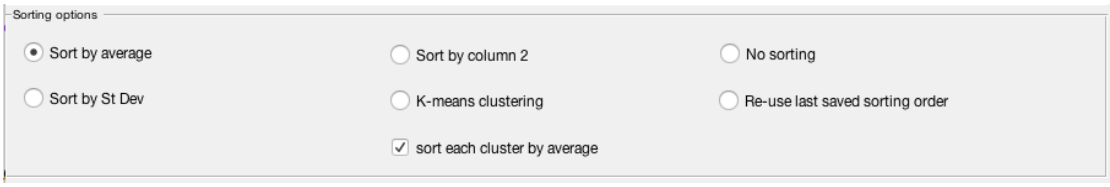
c.

shift=-1500, sort region=200



d.

Sorting options panel



The "Sorting options" panel allows to choose the way data will be sorted after normalization.

"Sort by average" – calculate the mean value in the +/- "sort region" for each row *Y* and sort the matrix accordingly

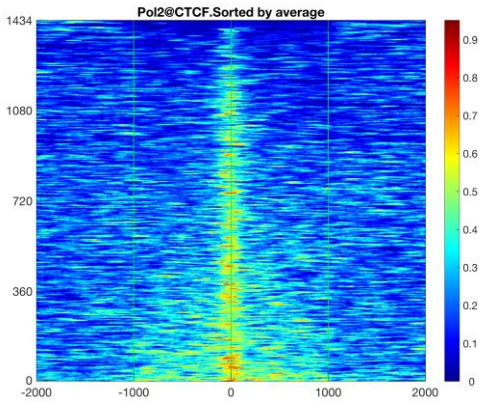
"Sort by St. Dev." – calculate the Standard Deviation value in the +/- "sort region" for each row *Y* and sort the matrix accordingly

"Sort by column 2" – Uses the column 2 of original input table for sorting. By default, the "Aggregate_profile.pl" script from the NucTools package output the occupancy matrix leaded by the transcripts length.

"No sorting" – Do not change sorting of the original matrix

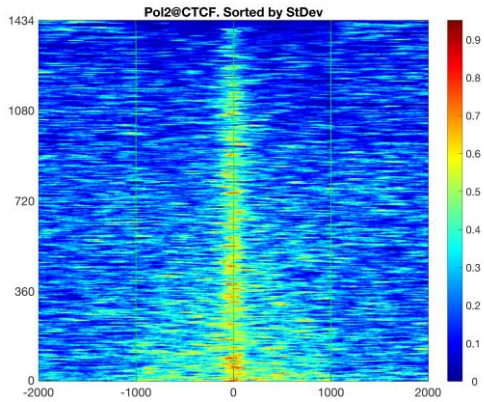
"K-means clustering" – perform K-mean clustering of the normalized/scaled data with the settings provided in the "K-means clustering settings" section.

"Sort by average"



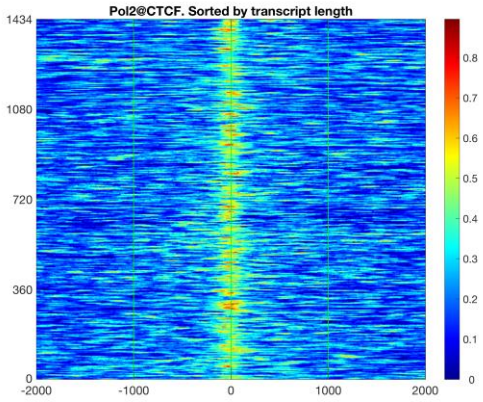
a.

"Sort by St. Dev."



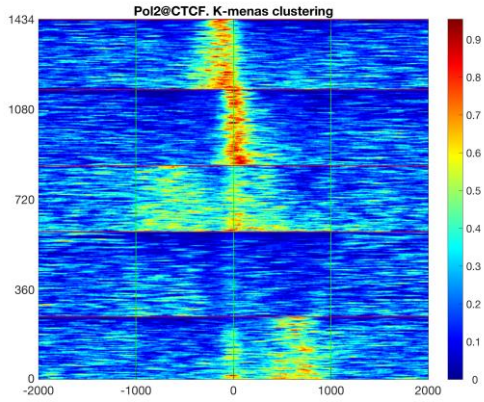
b.

"Sort by column 2"



c.

"K-means clustering"



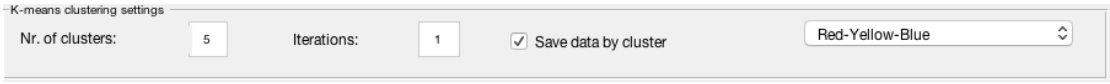
d.

Note: "Sort by average" and "Sort by ST. Dev." (panels a. and b.) options very often produce similar results with nucleosomes positioning data aligned at TF binding site, because the variability in the data for each feature directly connected to the number of reads
"Sort by column 2" in the panel c in this particular example is similar to a "no sorting" option, because original matrix does not carry transcript length information.

"Re-use last saved sorting order" – this sorting option allows applying clustering/sorting order achieved for one dataset to another dataset of the same size. This option is extremely useful when working with data series, for example

studying changes in nucleosome patterns in cells of healthy/diseased/treated patient, or differences in cell lines.
Every time one press "Start analysis" button new sorting order is created. Before application is closed, one can always reuse this sorting with the same matrix or apply to another matrix.

K-means clustering settings panel



"Nr.of clusters" – specify the number of expected clusters. The k-means clustering aims to partition all observations into k clusters in which each observation belongs to the cluster with the nearest mean.

"Iterations" – specify number of iterations for k-means algorithm. Increasing the number of iteration will produce more stable and reproducible clusters.

Note:

- *K-means algorithm is the simplest unsupervised learning algorithm and therefore always produces slightly different results because each time it starts from random assignment of clusters and further optimization for each data point. Nevertheless, most of the time the core of each identified clusters will be the same if algorithm converges.*
- *K-means clustering on the big data sets can be very time consuming. Minimizing the "sort region" allow to decrease significantly clustering performance*
- *Nr. of clusters parameter gives only approximate number. If the K-means algorithm can't converge with specified number of clusters, they will use lower number of clusters*
- *For more details about K-means clustering look for example here:* https://en.wikipedia.org/wiki/K-means_clustering

Save/Load sorting order panel

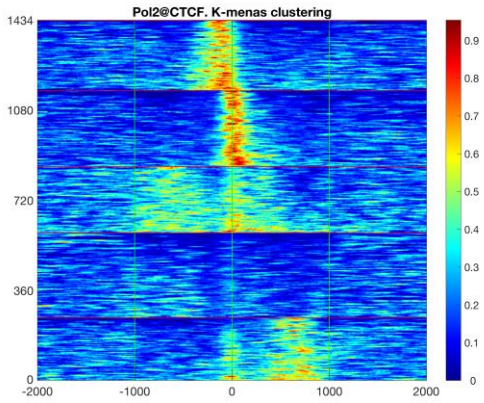
In order to preserve the sorting order for next analysis runs, one can save it to the file and load it back, pressing corresponding buttons.

The checkbox **"save sorting order"** should always be activate, in order to save the current analysis run.
"restore clusters" checkbox instructing the program to restore not only sorting, but as well the cluster order. If saved sorting order was derived from the un-clustered dataset, please disable this checkbox to avoid error message.

CMB graphical output

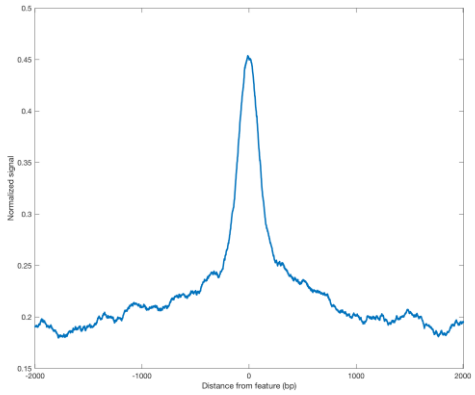
To illustrate the graphical output of CMB tool we are using the nucleosome density data around bound Pol2 in ESCs from low-MNase MNase-seq (Teif et al, 2014) around more than 100,000 sites of Pol2 enrichment in ESCs determined from ChIP-seq (mouse ENCODE). On the heat map, each horizontal line represents an individual genomic region containing Pol2 peak.
The typical CMB output consists of heatmap itself, average aggregated profile plot and aggregated plots for individual clusters:

Heatmap



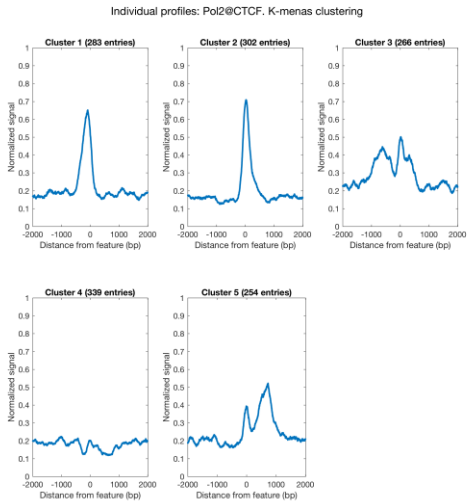
a.

Aggregated average profile



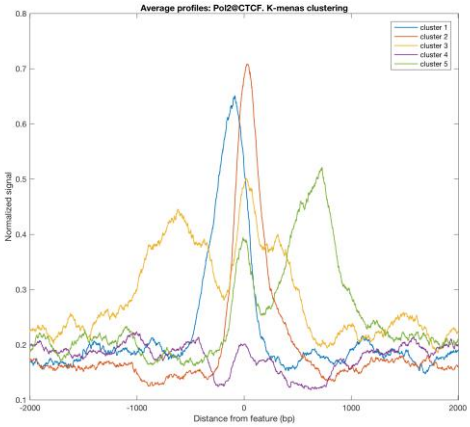
b.

Individual profiles (per cluster)



c.

All profiles (per cluster)



d.

 Y. Vainshtein; V. Teif

Known issues

Symptoms:
GUI starts properly, the data is loaded, but after pressing the "Start analysis" button appears an error message.

Reason:
Closing the CMB application by pressing window (x) button instead of "Exit" button sometimes causing the problem upon next start.
The default CMB settings file "settings.mat" is corrupted.

Solution 1:
After GUI appears, reactivate all options – double click all checkboxes, re-enter all numeric fields. After that close application with "Exit" button. After restarting a CMB tool everything should work fine.
Solution 2:
Close application, locate a "settings.mat" file in the CMB script directory and remove it. Restart the application.

Symptoms:
When loading the data table the error message "Wrong matrix file! Please use tab-delimited text tables with as minimum 2 columns and rows" pops-up.

Reason:
If you are loading 2D tab-delimited table with many columns but still see such message, the most probable reason is the wrong line endings in the text file (there is a operating system-specific difference in line ending of a text file)

Solution:

- Mac/Linux users: run a Perl one-liner replacing line endings in the Terminal session:

```
perl -pi -e 's/\r\n/\n/g' your_table.txt
perl -pi -e 's/\R/\n/g' your_table.txt
```

The command will replace original files with one with correct endings

- Windows users: load a table to Excel, remove last column and save the data using "Save As"->"Save as a Tab Delimited Text (*.txt)"